Erschienen in: Clematide, S; Klenner, M; Volk, M. Searching Answers -- Festschrift in Honour of Michael Hess on the Occasion of his 60th birthday. Münster, 85-99.

A Target-Driven Evaluation of Morphological Components for German

Cerstin Mahlow* and Michael Piotrowski**

Institute of Computational Linguistics
University of Zurich
Binzmühlestrasse 14, 8051 Zürich, Switzerland
{mahlow, mxp}@cl.uzh.ch

Abstract. In this paper we present an evaluation of rule-based morphological components for German for use in an interactive editing environment. The criteria for the evaluation are deduced from the intended use of these components, namely availability, performance, programming interfaces, and analysis quality. We evaluated systems developed and maintained since decades as well as new systems. However, we note serious general shortcomings when looking closer at recent implementations and come to the conclusion that the oldest system is the only one that satisfies our requirements.

1 Introduction

Today's computers have enough computational power to execute even complex natural language processing tasks fast enough to be integrated into real-world applications. In the *LingURed* project[1] we are implementing interactive editing functions to support writers during revising and editing, with the aim to reduce cognitive load and to prevent errors (see (Mahlow and Piotrowski, 2008; Piotrowski and Mahlow, 2009)). These functions are *language-aware*, i.e., they operate on linguistic elements and structures and respect the rules of a certain language. In the LingURed project we are concerned with language-aware functions for German; our target group are experienced writers (with respect to their knowledge of German, their writing, and their use of editors). Language-aware functions require linguistic knowledge and NLP resources for German.

* I came to Zurich in 2001 for a two-year e-learning project at the Institute of Computational Linguistics. The project is finished, the two years are over, but I still am with the institute as a guest, working on my PhD thesis for free and doing e-learning for other universities for money. Michael Hess always encouraged and supported me in aiming high and reaching these goals. He is my *Doktorvater* and always available for purely academic discussions. Happy Birthday, Michael!

** I would like thank Michael Hess for offering me a job that allowed me to join the first author in Zurich, and for creating a pleasant working environment in which I could finish writing my dissertation. He also showed me the beauty of *troff* and its use for advanced multi-channel publishing. Happy Birthday!

[1] LingURed stands for "Linguistically Supported Revising and Editing," see also `http://www.lingured.info/`.

The level of language dependence, as outlined in Mahlow et al. (2008), determines the kind of resource required for a certain editing function.

The quality of language-aware functions and their acceptance by users depends, on the one hand, on usability aspects, and on the other hand on the correctness of their results. The correctness of a function like `query-replace-word` relies on morphological components. This function replaces each occurrence of a word form of one word with the corresponding word form of another word, i.e., it takes into account the respective category, unlike usual "search & replace" functions. The better the quality of the results of the linguistic resource, the better the quality of the services built upon those resources. Therefore we make high demands on a morphological component.

In section 2 we outline the criteria for the evaluation and for the selection of the morphological systems for German to be evaluated. The results of our experiments are given in section 3.

2 System Requirements and Selection Criteria

2.1 System Requirements

An NLP resource has to meet certain requirements to be suitable for use in interactive, language-aware editing functions. The following list is an overview of key requirements for morphological components in the LingURed project. These requirements served as criteria in the evaluation described in section 3.

Availability For LingURed we use the XEmacs editor[2], a variant of Emacs (Stallman, 1981) as test bed. As XEmacs is open-source, all functions we implement should also be open-source. Thus, all required resources should also be freely available and redistributable.

Installation and Compilation To ensure off-line availability and optimal response time, the language-aware editing functions will not use Web services, but all needed resources and components should run locally on the writer's computer. The morphological component therefore should be portable and easy to install. If some form of compilation is required, the resources required in terms of time and computing power should be as small as possible.

Performance Since it will be used in an interactive environment, the morphological component has to start and execute quickly to ensure acceptable responsiveness. Users will not accept functions that make them wait more than a few seconds (less than two seconds for functions considered "easy" by the user) (see (Cooper et al., 2007; Good, 1981)).

Programming Interfaces and Further Processing The morphological component will take input from and deliver results to a calling Emacs Lisp function. It should therefore have programming interfaces to allow seamless integration, and the results delivered by the morphological component should be returned in a format suitable for further processing. The LingURed functions will not necessarily use all elements of the results, but it should be easy to access the information required by a

[2] `http://xemacs.org/`

particular function, e.g., only the lemma of an analyzed word form or the value for tense is needed. To generate a certain word form or the paradigm of a word, as few parameters as possible should be required.

Quality of the Results The morphological component should have a high coverage. Analyses should be complete and correct. Hausser (2001) formulates two main requirements for automatic morphological analysis: Each input word form "must be characterized automatically with respect to categorization and lemmatization", where *categorization* consists in "specifying the part of speech [...] and the morphosyntactic properties of the surface", and *lemmatization* consists in "relating a word form [...] to the corresponding base form" (Hausser, 2001, 251f.).[3] These requirements seem obvious, but we will see that they are only met by some of the current morphological systems.

For generation, it should be possible to generate a concrete word form by calling the morphological component with the lemma and the desired category. Additionally, we expect the component to be able to generate the paradigm of a certain word. As a specific requirement for German, for verbs with separable prefixes we would expect both the forms used in main clauses (i.e., with separated prefix like *Er **schreibt** es **auf**.* 'He writes it down.') and those used in subordinate clauses (i.e., written as one word, like *[...], damit er es **aufschreibt**.*) or equivalent information about separable prefixes.

Furthermore, we generally expect components to deliver high-quality results without any need for post-processing to correct errors.

2.2 Selection of Systems for Evaluation

As in many areas of computational linguistics, there are rule-based and statistical approaches to morphological analysis. In the evaluation described in this paper, we have only considered rule-based systems.

One factor was our own experience in implementing rule-based morphological analyzers, so we know that they are able to deliver detailed, structured analyses (see Mahlow and Piotrowski (2009)). The possibility to draw upon the morphological processes of inflection, derivation, and compounding involved when analyzing or generating word forms, the respective parse and generation trees, and certain elements of the category, is, in our view, a potential advantage when compared to statistically created results.

The second and deciding factor is that, based on the results of Morpho Challenge[4], we have come to the conclusion that statistical morphological analyzers are not yet able to deliver the quality of results we require. In the Morpho Challenge morpheme analysis task, the analyses proposed by the participants' algorithms are compared against a linguistic gold standard. At Morpho Challenge 2008 (Kurimo and Varjokallio, 2008), the best system for German achieved an F-measure of 54.06%. The best recall value

[3] We refer to "morphosyntactic properties" as "category," while the process of determining this category is called "categorization."

[4] Morpho Challenge is a shared task and conference for the evaluation of statistical morphological components based on unsupervised machine-learning. The morpheme analysis task ("Competition 1") is conducted for Arabic, English, Finnish, German, and Turkish.

was 59.51% (this system achieved 49.53% precision), the best result for precision was 87.92% (with 7.44% recall).[5] These figures are much too low to consider the systems as suitable for use in interactive applications.

We were able to obtain the following four systems for evaluation: (1) Stripey Zebra (Lorenz, 1996; Schulze, 2004), (2) Morphisto (Zielinski and Simon, 2008), (3) GERTWOL (Koskenniemi and Haapalainen, 1996), and (4) mOLIFde (Clematide, 2008). Further general information on these systems is given in section 3.1.

3 Experiments

3.1 Availability and Installation

Stripey Zebra is the current version of the German morphology originally developed as DMM at the University of Erlangen since the mid-1990s. It is based on the Malaga framework[6], which implements the *Left-associative Grammar* (LAG) formalism (Hausser, 2001).

For our experiments we used Malaga 7.12, freely available under the GNU General Public License (GPL). Stripey Zebra itself, i.e., the lexicon and grammar rules, has to be licensed from the developers; we used version 1.1. Stripey Zebra is platform-independent and requires only the Malaga virtual machine; we have installed this and earlier versions of Malaga using the familiar "`configure`, `make`, `make install`" sequence on various versions of Solaris, HP-UX, NetBSD, Mac OS X, and Linux. The installation from source takes about 10 minutes. The lexicon contains about 50200 baseform entries.[7] Stripey Zebra is shipped in compiled form and can be used immediately. The rules and the lexicon are thus not modifiable, but additional lexicon entries can be added to a user lexicon.

Morphisto was developed at the Institut für Deutsche Sprache (IDS), Mannheim, within the TextGrid project[8] and is thus one of the most recent developments in the field of morphological components for German. It is in itself not a complete morphology, but rather an open-source lexicon and a set of patches for the SMOR morphology (Schmid et al., 2004). SMOR is based on the Stuttgart Finite State Transducer Tool (SFST)[9]. For simplicity, when we refer to Morphisto, we mean the combination of SFST, SMOR, and the Morphisto lexicon.

For our experiments we used SFST 1.3, which is freely available under the GPL, and the version of SMOR which is included in the SFST distribution. We have built SFST on Mac OS X, NetBSD, and Linux systems. We then installed the Morphisto release[10] from December 12, 2008, also freely available under a Creative Commons license, which

[5] See `http://www.cis.hut.fi/morphochallenge2008/` for details. The results of Morpho Challenge 2009 have not yet been published.

[6] `http://home.arcor.de/bjoern-beutel/malaga/`

[7] This number is based on the DMM version 5.0, the predecessor of Stripey Zebra 1.1.

[8] `http://www.textgrid.de/`

[9] `http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/SFST.html`

[10] Available from `http://ids-mannheim.de/ll/TextGrid/morphisto.html`

patches the SMOR rules and, most importantly, replaces the included toy lexicon with a comprehensive lexicon containing 19234 entries. The lexicon is distributed in source form and must be compiled before use. The Morphisto Web site states that *at least* 8 GB of RAM are required for compiling the transducer—we can confirm this requirement: The compilation took more than two and a half hours on our server[11]. The compilation of the inverted transducer (for generation) took the same amount of time. Compacting the transducer for faster execution took six seconds. Thus, it takes more than five hours and a computer with sufficient RAM to compile the whole component. The compiled transducers (each 23 GB) can then also be used on systems with less resources.

GERTWOL (and the generator component GERGEN) are a commercial product[12] developed and distributed by Lingsoft Oy[13], Helsinki, since the mid-1980s. It is only available in binary form for several platforms, including Mac OS X and Linux. It is based on finite-state technology and Koskenniemi's two-level approach to morphology (Koskenniemi, 1983). An earlier version of GERTWOL was the winner of the First Morpholympics competition in 1995 (Hausser, 1996). GERTWOL is primarily designed for use in spelling checkers; Lingsoft produces the proofing tools for German included in Microsoft Office.

For our experiments we used the GERTWOL release from April 22, 2009 with the LSINDEX API. Lingsoft ships a shared library (likely to contain a version of the Xerox finite-state engine) and platform-independent lexicon files. The current lexicon is claimed to contain over 150000 entries. According to the documentation, there are various versions available, which cover different spelling conventions. The version available to us by a license is apparently the version for the 2006 spelling for all German-speaking countries except Switzerland. For the evaluation, we have written a small program which reads word forms from standard input and prints the analyses to standard output.

mOLIFde is an experimental morphology, which is under development at the University of Zurich. It is based on the Xerox Finite-State Tool (XFST)[14] and a lexicon in the Open Lexicon Interchange Format (OLIF)[15]. The mOLIFde lexicon and rules are planned to be made available under an open-source license, but they have not yet been published at the time of this writing.

XFST must be licensed from Xerox; when buying Beesley and Karttunen (2003), one receives a license for XFST, further updates are then free. XFST is distributed in binary form for several platforms. We have successfully installed XFST on Mac OS X, Solaris, and Linux. The `lookup` utility of the XFST distribution we used for our experiments identified itself as version 2.3.0 (8.0.5). Compiling the mOLIFde transducer

[11] 2 2.5 GHz Dual Core Opteron processors, 8 GB RAM, running Ubuntu 8.04.

[12] Special licenses for academic research are available.

[13] `http://lingsoft.fi/`

[14] See Lauri Karttunen, Tamás Gaál, and André Kempe, *Xerox Finite-State Tool*, available online: `http://www.cis.upenn.edu/~cis639/docs/xfst.html`, January 12, 1998, last visited on August 31st, 2009.

[15] `http://www.olif.net/`

from a lexicon with 43677 lemma entries[16] took ten minutes on a MacBook (2.16 GHz Intel Core 2 Duo processor, 2 GB RAM, running Mac OS X 10.5.4). The transducer can then be used for analyzing and generating.

Summary All four systems are available for a variety of platforms, which is important for LingURed, as the morphological analysis is to take place on the writer's computer. However, there is only one system which is completely open-source, namely Morphisto, but compiling it requires significant hardware resources.

3.2 Performance

To get an impression of the performance of the four systems we conducted small experiments. We had each system analyze two small corpora:

1. The text of the April 1994 issues of the *Neue Zürcher Zeitung* (NZZ) newspaper, consisting of 324463 running word forms and 54805 unique word forms. The NZZ corpus is marked by the use of Swiss-German spelling (which doesn't use "ß") and Helvetisms.
2. The first 325000 running word forms from the Limas corpus[17], with 52513 unique word forms. The texts of the Limas corpus are mainly from the 1970s and are written in pre-1996 orthography.

For the performance evaluation, we were interested in (1) the ratio of unrecognized to recognized word forms, (2) the number of analyses per word form, and (3) the time needed to analyze the complete corpora and the lists of unique word forms, and thus the number of word forms analyzed per second. The correctness of the analyses was evaluated in a separate experiment described in section 3.4.

Tables 1 and 2 show the results of the performance tests. All tests were run on the MacBook described above. For Stripey Zebra we used two settings: Once with the "weighting" and "robust" features enabled (only the most likely analyses are returned and hypotheses are generated for unknown words, which results in the 100% analysis rate), and once with both features disabled.

There are two peculiarities in the results: The extremely low percentage of word forms analyzed by mOLIFde and the very low number of analyses per analyzed word form delivered by Stripey Zebra.

One explanation for the low number of results for mOLIFde is the fact that it handles only verbs, nouns, and adjectives, and that the system is still under development. As a coverage of less than 50% is much too low for real-world applications, mOLIFde was excluded from the experiments described in the following sections; note that the goal of the experiments described here was to find the best morphological component for use in the LingURed project, not a general evaluation.

[16] As of June 1, 2009.

[17] `http://www.korpora.org/Limas/`

Table 1. Performance on the NZZ corpus.[18]

	324463 running word forms					54805 unique word forms				
	WF Anlyzd.	% Anlyzd.	ANA per WF	Time (s)	WF/s	Anlyzd. WF	% Recog.	ANA per WF	Time (s)	WF/s
GERTWOL	305208	94.07	6.02	117	2773	46130	84.17	6.82	27	2030
Morphisto	292533	90.16	5.91	13	24958	42511	77.57	9.30	4	13701
SZ w/o wt.	299759	92.39	1.87	126	2575	43996	80.28	2.33	39	1405
SZ w/ wt.	324463	100.00	1.00	143	2268	54805	100.00	1.00	44	1245
mOLIFde	98844	30.46	4.86	20	16223	13530	24.68	5.73	5	10964

Table 2. Performance on the first part of the Limas corpus.[18]

	325000 running word forms					52513 unique word forms				
	WF Anlyzd.	% Anlyzd.	ANA per WF	Time (s)	WF/s	Anlyzd. WF	% Recog.	ANA per WF	Time (s)	WF/s
GERTWOL	312675	96.21	5.86	117	2777	46947	89.40	6.79	28	1875
Morphisto	306798	94.40	5.75	13	25000	43997	83.78	9.52	4	13128
SZ w/o wt.	314246	96.69	1.91	133	2443	46314	88.20	2.47	44	1193
SZ w/ wt.	325000	100.00	1.00	144	2256	52513	100.00	1.00	47	1117
mOLIFde	102081	31.41	4.64	15	21666	13530	32.81	5.73	5	10964

The small number of analyses per analyzed word form delivered by Stripey Zebra can be explained by its design. "Weighting" means that only the most probable analyses will be returned, a feature not available in the other systems. Furthermore, Stripey Zebra uses a so-called *distinctive* categorization scheme, while the other systems use *exhaustive* categorization[19].

The overall impression of Stripey Zebra, Morphisto and GERTWOL is good. The numbers for GERTWOL are comparable to the performance of the older version that participated in the Morpholoympics. Morphisto is faster than Stripey Zebra and GERTWOL, but recognizes a lower percentage of word forms.

3.3 Programming Interfaces and Further Processing of Results

To make practical use of morphological analysis and generation results in applications, it is critical that applications can integrate the morphological component and receive the

[18] Abbreviations: *WF Anlyzd.*: Number of word forms analyzed; *% Anlyzd.*: Percentage of word forms analyzed; *ANA per WF*: Average number of analyses per analyzed word form; *Time (s)*: Time needed for the analysis, in seconds, *WF/s*: Word forms analyzed per second. *SZ w/o wt.*: Stripey Zebra without the "weighting" and "robust" features; *SZ w/ wt.* Stripey Zebra with these features enabled.

[19] See (Hausser, 2001, pp. 244, 346) for the differences between distinctive and exhaustive categorization.

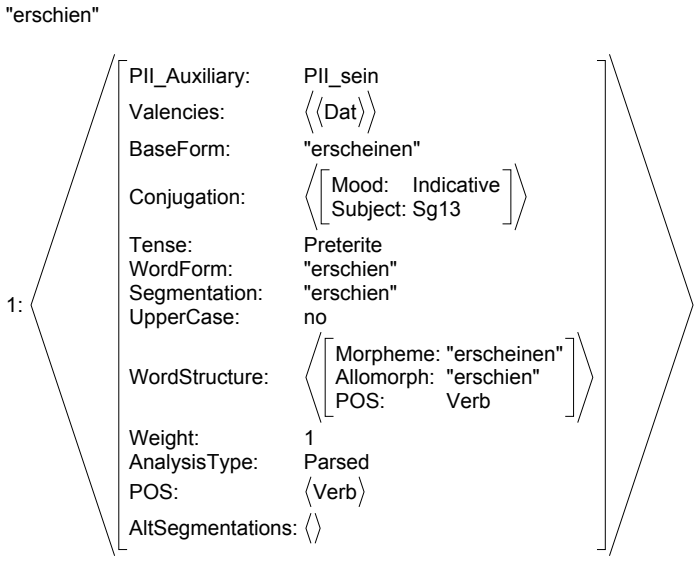

Fig. 1. Analysis of *erschien* by Stripey Zebra

results in a format suitable for further processing. For illustration we show the analysis for the word form *erschien* 'appeared' delivered by each system.

For Stripey Zebra, Malaga provides interactive and batch-mode utilities and a C library and API for this purpose, and there are modules for Perl, Ruby, and Python, which allow convenient processing of analysis results. The analyses are represented by nested feature-value structures as shown in figure 1. Note that Malaga can also output the results as text, or they can be accessed via the API.

For Morphisto, SFST provides interactive and batch-mode utilities and a C++ library; there is also a Python module available. The analysis of a word form consists of a list of readings and the associated categories. A category is a string like `V 3 Sg Pres Ind`, so it is not possible to directly request the value for, e.g., part of speech. See listing 1 for an example of an analysis.

```
> erschien
erscheinen<+V><1><Sg><Past><Ind>
erscheinen<+V><3><Sg><Past><Ind>
```

Listing 1. Analysis of *erschien* by Morphisto

GERTWOL is shipped as shared library with a C API for integration into applications; there is a demonstration program, but no standard utilities as for SFST. The user thus has to write a program to make use of GERTWOL. In GERTWOL analyses the category is represented as an array of "tags," where each array position corresponds to a grammatical feature, e.g., position 1 contains the part of speech and position 6 contains the case or is empty, if not applicable. Listing 2 shows the information returned by GERTWOL; empty tags are replaced by "–".

```
"<erschien>"
    "er|schein~en"  V - - - - - - - PAST IND - - - SG1 -
    "er|schein~en"  V - - - - - - - PAST IND - - - SG3 -
    "er|schien~en"  V - - - - - - - PRES IMP - - - SG2 -
```

Listing 2. Analysis of *erschien* by GERTWOL

3.4 Quality of the Analyses

We tested the three remaining systems, Stripey Zebra, Morphisto, and GERTWOL with respect to the quality of the analyses they deliver. In this section, we describe our evaluation methodology and the results. The systems' capabilities for generating word forms and paradigms are briefly discussed in section 3.5.

The First Morpholympics in 1994 were the first and, up to now, the only large-scale competition of morphological systems for German. Unfortunately, the *correctness* of the analysis results of the participating systems was not evaluated in the Morpholympics (Lenders et al., 1996, p. 14), so that no comparison data is available.

As can be seen from tables 1 and 2, the number of recognized word forms for Stripey Zebra, Morphisto, and GERTWOL is above 90% for running word forms and above 80% for unique word forms. The number of analyses per recognized word form differs between the systems; the use of distinctive categories by Stripey Zebra and exhaustive categories by the other systems explains only some of these differences.

The first question with respect to the quality of analysis is: Given some German text, how many of the word forms will be analyzed correctly, where *correct* means, that *all* analyses for a certain word form are correct and no analysis is missing. To be able to give a general statement, we randomly chose 384 word forms from each of the NZZ and Limas corpora. The sample size was chosen to achieve a confidence level of 95% with a 5% error, according to the standard formula

$$n = \frac{Z^2\sigma^2}{e^2} \tag{1}$$

where $Z^2 = 1.96$ for a confidence level of 95%, e is the desired level of precision (we use a confidence interval of 5%), and σ^2 is the variance of an attribute in the population (we assume $\sigma^2 = .25$ for maximum variability).

These two sets of 384 word forms were analyzed by each system and manually evaluated by a single annotator for correctness.[20] The results are shown in figure 2. The number of "recognized" word forms (80 to 90%) does not correspond with the number of word forms analyzed completely correctly (around 60%). However, it can be said that Stripey Zebra and GERTWOL are very similar, and both are better than Morphisto. An impression that is consistent with the performance tests.

A closer look at the results uncovers more differences between the systems. Figure 3 shows the detailed results of the evaluation. The analyses of the word forms were rated on the following scale: (1) all and only correct analyses[21] for the word form are returned,

[20] The Limas sample contained 9 misspelled word forms, which were excluded from the evaluation.

[21] *Correct* means: At least the correct lemma and the correct category were produced; segmentation, valencies, and other additional information were not considered.

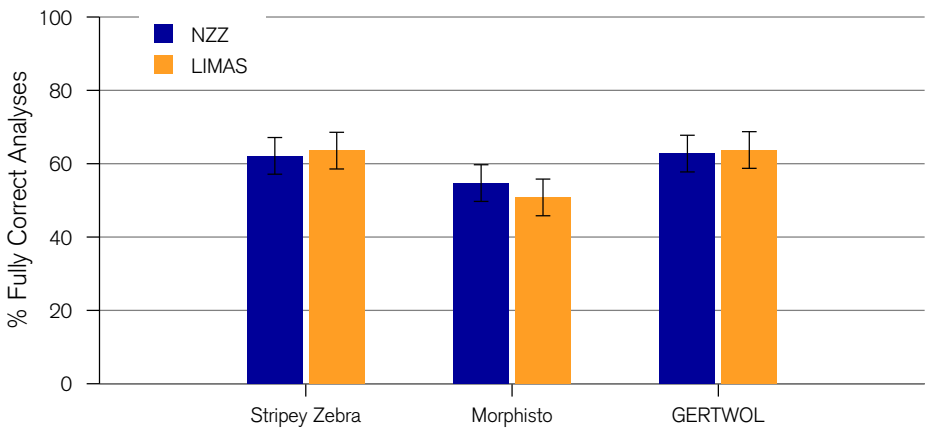

Fig. 2. Percentage of completely correct analyses with a confidence interval of 5%

(2) only correct analyses are returned, but they are incomplete, (3) all correct, but also some incorrect analyses are returned, (4) some correct and some incorrect analyses are returned, (5) only incorrect analyses are returned, and (6) no analysis is returned. We now discuss the results for each system in detail.

Stripey Zebra We ran Stripey Zebra with weighting enabled, but without robust rules, i.e., no hypotheses were generated for unknown words. A high percentage of analyses was rated 2 (only correct analyses are returned, but they are incomplete). The main reason for this is the format of the analyses for inflected adjectives which contain the value `DeclAdjective` for `POS` and a tag as value for `AdjDeclension`. There is no detailed information on gender, case, number, and whether it is of strong, weak or mixed declension. These values could be inferred from the `AdjDeclension` tag, but are not included as such in the analysis, which was therefore rated 2.[22] There are almost no analyses rated 3, 4, or 5. Fixing the problem in the analyses of adjectives would thus increase the number of completely correct analyses, which could then approach the number of recognized word forms.

Morphisto The percentage of analyses rated 2 is very low. There is a high number of analyses rated 3 and 5. Most of these (partly) incorrect analyses are due to the fact that Morphisto—or rather the SMOR morphology—fails to deliver a lemma. The analyses of Morphisto consist of a category and a string that looks like the lemma. However, when analyzing compounds or derived word forms, it becomes obvious that this string in fact shows the segmentation of a word form using the involved base forms, not the allomorphs. Furthermore, linking morphemes (*Fugenelemente*) are missing. It is therefore not possible to infer the lemma for words such as *Vermittlerrolle* 'role as

[22] This seems to be a hard penalty, since post-processing the results can be done easily. However, our requirements in section 2.1 clearly demand exact results without post-processing.

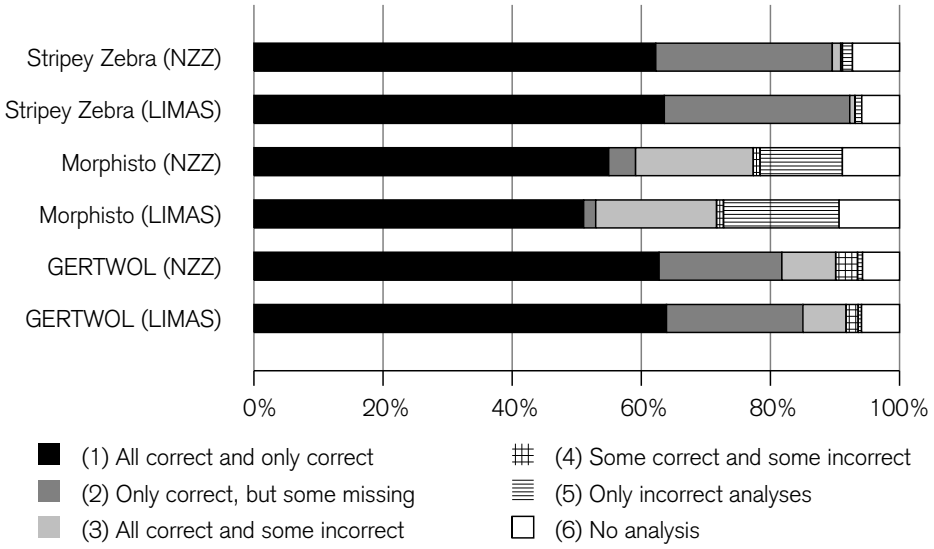

Fig. 3. Classification and percentages of analyses

mediator' or *Jahrespressekonferenz* 'anual press conference', for which one gets analyses like the following:

```
vermitteln<V>er<NN><SUFF>Rolle: NN Fem Nom Sg
Jahr<NN>Presse<NN>Konferenz: NN Fem Nom Sg
```

Unlike the shortcomings of Stripey Zebra, which could be easily corrected by expanding the tags, the extraction of the lemma from these strings would require linguistic knowledge and cannot be solved with a simple table lookup.

Thus, surprisingly, Morphisto, which is described as *lemmatizer*, is unable to deliver the lemma for a word form and thus fails to complete one of the main tasks of automatic morphological analysis. In the light of this, none of the results delivered by Morphisto should have been categorized as completely correct, but we were more lenient and treated the string one gets by stripping the tags in angle brackets as the lemma and then checked whether it would be the correct one.

GERTWOL As for Stripey Zebra, a relatively high percentage of results were rated 2 (only correct analyses are returned, but they are incomplete), which is also caused by problems in the analysis of inflected adjectives. GERTWOL returns explicit categories for inflected adjectives, but it does not take into account the class of mixed declension. Thus, all word forms of adjectives that could also be a form of the mixed declension are missing from these analyses. As it is very regular, this problem could certainly be solved by extending the system. However, since source code access is not available, one could only use some form of post-processing.[23] GERTWOL combines the lemma and

[23] As for Stripey Zebra (see above), we decided to be very strict in categorizing the analyses.

the segmentation in a single string. In contrast to Morphisto, however, the boundary characters from this string can be suppressed in the C API using the shared library to obtain the actual lemma.

Summary The quality of the analyses delivered by Stripey Zebra and GERTWOL could be improved easily—if the systems were open source. Since both systems are available with no access to the sources, we would have to contact the developers and ask for improvement[24], or we would have to do some post-processing. If we expect that most of the analyses rated 2 without post-processing could be rated 1, Stripey Zebra would deliver around 90% completely correct analyses, and GERTWOL would deliver around 82% completely correct analyses. Thus, if Stripey Zebra delivers an analysis for a word form, this analysis would almost always be completely correct. The shortcomings of Morphisto cannot be resolved with simple post processing but would involve further development—not only for the problem of the missing lemma, but also for otherwise incorrect or missing analyses.

3.5 Generation

In this evaluation, we have only evaluated the quality of the morphological analyses. An evaluation of the generation quality proved to be pointless due to the fact that only one system is usable for generation.

As Malaga does not support generation, it is not possible to generate word forms or paradigms of words using Stripey Zebra. The workarounds described by Mahlow and Piotrowski (2009) would either require source code access to the grammar rules or are too slow for use in interactive settings.

Generation with Morphisto is possible, but its practical use is severely hampered by the lack of real lemmatization, which makes it impossible to analyze a word form of a compound word and to generate a different word form of this word.

The generation version of GERTWOL is called GERGEN. GERGEN is effectively the only usable generation component. Neither Morphisto nor GERGEN offer a function for generating the paradigm of a word; in the case of GERGEN this is despite the documentation stating the opposite. It is thus necessary to code the required linguistic information in some other location and then call the generation several times to produce all required word forms. Also, both systems only generate the unseparated form of verbs with separable prefixes, which means that the application also needs to know about separable prefixes.

4 Summary and Conclusion

In this paper we have presented an evaluation of four current rule-based systems for morphological analysis and generation of German word forms. We compared Stripey Zebra, Morphisto, GERTWOL, and mOLIFde with respect to availability, performance,

[24] However, since both systems were developed originally with a certain purpose, the developers might consider the shortcomings not as bugs but as features.

Table 3. Overview of the results of the evaluation. Criteria marked with * are considered critical, whereas other criteria are considered as "nice to have," but may still influence the final decision.

Criteria	Stripey Zebra	Morphisto	GERTWOL	mOLIFde
Open Source	±	+	−	±
Easy to install/compile	+	−	+	+
Speed	+	+	+	+
Coverage *	+	+	+	−
Interfaces *	+	±	±	n/a
Further processing *	+	±	+	n/a
Quality of analyses *	+	±	+	n/a
Generation *	−	±	+	n/a

embeddability into applications, and suitability of the analyses for further processing. We conducted a small-scale experiment to determine the quality of the analyses delivered by the first three systems. mOLIFde is still under development and does not yet achieve comparable performance. We performed this evaluation to determine the morphological system best suited for use as an NLP component for interactive language-aware editing functions in the LingURed project.

Table 3 summarizes the results of the experiments. mOLIFde was not considered for further evaluation after the performance tests.

The results prompted us to select GERTWOL. It is, in fact, the only system suitable for our application at all. Although Morphisto and mOLIFde are fast and (mainly) open-source, they showed severe shortcomings with respect to the analyses they deliver. Stripey Zebra and GERTWOL are almost equal in terms of performance and analysis quality; both of them are not freely available but require a license. In the end, the deciding factor was the ability of GERTWOL to also generate word forms; while Stripey Zebra delivered more detailed analyses, the underlying framework is not suitable for generation.

Interestingly, the oldest system turned out to best meet our requirements. GERTWOL is also the only system developed by non-native speakers. Reasoning why the quality of GERTWOL—which had also won the Morpholympics in 1994—still outperforms all other systems is left to the reader: Are native speakers somehow "blinded by routine", or does it perhaps require *sisu*[25] to implement a good rule-based morphological system for German? In any case, the current state of the art in morphological analysis and generation for German is disappointing. It is hard to see any improvement as compared to 1994.

The criteria of this evaluation were oriented towards finding the best morphological component to be used in interactive editing functions. While they may also give an indication on the systems' performance in other applications, further experiments are

[25] The untranslatable characteristic that only Finns possess, which may roughly be described as strength of will, determination, perseverance, and rational acting in the face of adversity.

necessary to give a more comprehensive assessment of the quality of today's rule-based morphological components for German. In particular, an evaluation against a pre-determined gold standard, as in Morpho Challenge, would be interesting.

References

Beesley, K. R. and Karttunen, L. (2003). *Finite State Morphology*. Center for the Study of Language and Information.

Clematide, S. (2008). An OLIF-based Open Inflectional Resource and yet another Morphological System for German. In Storrer, A., Geyken, A., Siebert, A., and Würzner, K.-M., editors, *Text Resources and Lexical Knowledge: Selected Papers from the 9th Conference on Natural Language Processing KONVENS 2008*, pages 183–194. Mouton de Gruyter.

Cooper, A., Reimann, R., and Cronin, D. (2007). *About Face 3: The Essentials of Interaction Design*. Wiley, Indianapolis, IN, USA.

Good, M. (1981). Etude and the folklore of user interface design. In *Proceedings of the ACM SIGPLAN SIGOA symposium on Text manipulation*, pages 34–43, New York, NY, USA. ACM.

Hausser, R. (1996). *Linguistische Verifikation. Dokumentation zur Ersten Morpholympics*. Niemeyer, Tübingen.

Hausser, R. (2001). *Foundations of Computational Linguistics: Human-Computer Communication in Natural Language*. Springer, 2nd rev. and ext. edition.

Koskenniemi, K. (1983). *Two-level Morphology: A General Computational Model for Word-Form Recognition and Production*. PhD thesis, University of Helsinki.

Koskenniemi, K. and Haapalainen, M. (1996). GERTWOL – Lingsoft Oy. In Hausser, R., editor, *Linguistische Verifikation. Dokumentation zur Ersten Morpholympics 1994*, chapter 11, pages 121–140. Niemeyer, Tübingen.

Kurimo, M. and Varjokallio, M. (2008). Unsupervised morpheme analysis evaluation by a comparison to a linguistic gold standard – Morpho Challenge 2008. In *Workshop of the Cross-Language Evaluation Forum, CLEF 2008*.

Lenders, W., Bátori, I., Dogil, G., Görz, G., and Seewald, U. (1996). Stellungnahme der Jury für die Morpholympics 94. In Hausser, R., editor, *Linguistische Verifikation. Dokumentation zur Ersten Morpholympics 1994*, pages 15–24. Niemeyer, Tübingen.

Lorenz, O. (1996). Automatische Wortformerkennung für das Deutsche im Rahmen von MALAGA. Master's thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg.

Mahlow, C. and Piotrowski, M. (2008). Linguistic support for revising and editing. In Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing: 9th International Conference, CICLing 2008, Haifa, Israel, February 17–23, 2008. Proceedings*, pages 631–642, Heidelberg. Springer.

Mahlow, C. and Piotrowski, M. (2009). SMM: Detailed, structured morphological analysis for Spanish. *Polibits. Computer science and computer engineering with applications*, (39):41–48.

Mahlow, C., Piotrowski, M., and Hess, M. (2008). Language-aware text editing. In Dale, R., Max, A., and Zock, M., editors, *LREC 2008 Workshop on NLP Resources, Algorithms and Tools for Authoring Aids*, pages 9–13, Marrakech, Morrocco. ELRA.

Piotrowski, M. and Mahlow, C. (2009). Linguistic editing support. In *DocEng'09: Proceedings of the 2009 ACM Symposium on Document Engineering*, pages 214–217, New York, NY, USA. ACM.

Schmid, H., Fitschen, A., and Heid, U. (2004). A german computational morphology covering derivation, composition, and inflection. In *IVth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1263–1266.

Schulze, M. (2004). *Ein sprachunabhängiger Ansatz zur Entwicklung deklarativer, robuster LA-Grammatiken mit einer exemplarischen Anwendung auf das Deutsche und das Englische*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg.

Stallman, R. M. (1981). EMACS the extensible, customizable self-documenting display editor. In *Proceedings of the ACM SIGPLAN SIGOA symposium on Text manipulation*, pages 147–156, New York, NY, USA. ACM.

Zielinski, A. and Simon, C. (2008). Morphisto: An Open-Source Morphological Analyzer for German. In *Seventh International Workshop on Finite-State Methods and Natural Language Processing*, pages 177–184.